

学校编码: 10384

分类号\_\_\_\_密级\_\_\_\_

学号: X2009222002

UDC\_\_\_\_\_

厦 门 大 学

工 程 硕 士 学 位 论 文

基于 LEON3 的 G.723.1 声码器 SOC 软硬件协同设计  
与实现研究

SW&HW Co-Design and Implementation Research of  
G.723.1 Vocoder SOC based on LEON3

郭子超

指导教师姓名: 周剑扬 副教授

专 业 名 称: 电子与通信工程专业

论文提交日期: 2012 年 6 月

论文答辩时间: 2012 年 6 月

学位授予日期: 2012 年 月

答辩委员会主席: \_\_\_\_\_

评 阅 人: \_\_\_\_\_

2012 年 6 月

厦门大学博硕士论文摘要库

## 厦门大学学位论文原创性声明

本人呈交的学位论文是本人在导师指导下，独立完成的研究成果。本人在论文写作中参考其他个人或集体已经发表的研究成果，均在文中以适当方式明确标明，并符合法律规范和《厦门大学研究生学术活动规范（试行）》。

另外，该学位论文为（  
）课题（组）  
的研究成果，获得（  
）课题（组）经费或实验室的  
资助，在（  
）实验室完成。（请在以上括号内填写课题或课题组负责人或实验室名称，未有此项声明内容的，可以不作特别声明。）

声明人（签名）：

年 月 日

厦门大学博硕士论文摘要库

## 厦门大学学位论文著作权使用声明

本人同意厦门大学根据《中华人民共和国学位条例暂行实施办法》等规定保留和使用此学位论文，并向主管部门或其指定机构送交学位论文（包括纸质版和电子版），允许学位论文进入厦门大学图书馆及其数据库被查阅、借阅。本人同意厦门大学将学位论文加入全国博士、硕士学位论文共建单位数据库进行检索，将学位论文的标题和摘要汇编出版，采用影印、缩印或者其它方式合理复制学位论文。

本学位论文属于：

（        ） 1. 经厦门大学保密委员会审查核定的保密学位论文，  
于        年        月        日解密，解密后适用上述授权。

（        ） 2. 不保密，适用上述授权。

（请在以上相应括号内打“√”或填上相应内容。保密学位论文应是已经厦门大学保密委员会审定过的学位论文，未经厦门大学保密委员会审定的学位论文均为公开学位论文。此声明栏不填写的，默认为公开学位论文，均适用上述授权。）

声明人（签名）：

年        月        日

厦门大学博硕士论文摘要库

## 摘 要

随着开源硬件的蓬勃发展，开源处理器软核越来越被学术界和工业界接受和使用。特别是 Aeroflex Gaisler 公司开发的 LEON 系列开源软核处理器，已经成为研究和应用的热点。基于这个背景，本文以 G723.1 声码器的 SOC 实现作为案例，研究探讨了基于 LEON3/GRLIB 平台的 SOC 设计和验证方法，展示了程序性能分析-硬件加速-软件优化三部曲的软硬件协同设计过程。由于 LEON3 官方文档中缺失对于协处理器的描述，另外 VHDL 源代码对于协处理器的支持也不完善，本文的最大贡献在于重点分析了 LEON3 的协处理器接口并详细描述了协处理器的设计实现细节。

本文首先介绍了基于 LEON3/GRLIB 平台的 SOC 架构和仿真/验证方法，然后对 ITU-T G723.1 参考程序的性能进行了测量和剖析，确定了硬件加速的目标。接下来在重点篇章里，详细阐述了在 LEON3 上实现一个流水线的协处理器的设计方法和针对 G723.1 算法加速的协处理器的设计细节。面对会严重降低指令性能的流水线数据冲突和指令操作数位宽问题，对不同解决方法进行深入的理论分析之后采用了数据前馈和镜像寄存器的技术来解决。在硬件实现之后，使用自定义的协处理器指令对算法程序进行优化并分析优化的结果。此外还对 SOC 的存储子系统配置做了深入的分析 and 实验，得出了理论和实际一致的结论。最后 G723.1 算法程序在 FPGA 原型上的运行结果证明了协处理器设计的正确和有效性。

**关键词：** LEON3；协处理器；G723.1

厦门大学博硕士论文摘要库



## Abstract

As the rapid development of open-sourced HW, open-sourced soft-processors become more and more accepted and used by academic circles and industrial circles. Especially the open-sourced LEON-series processors developed by Aeroflex Gaisler Inc., has been extensively researched and exploited. In such a situation this thesis illustrates an implementation case of G723.1 vocoder, which demonstrates methodologies of SOC design and verification based on LEON3/GRLIB SOC platform. And it presents a SW & HW co-design flow in combination of SW performance profiling, HW accelerating and SW optimization. Due to lack of coprocessor description in LEON3 official documents, and incomplete coprocessor support in LEON3 VHDL model, the most significant contribution of the thesis is analysing LEON3's coprocessor interface and describing in detail the design and implementation of coprocessor.

The thesis first introduces the SOC architecture based on LEON3/GRLIB platform. Secondly the performance of ITU-T G723.1 reference code is measured and profiled, which establishes the objective of HW acceleration. Next is the most important chapter of the thesis, which explains how to implement a pipelined coprocessor on LEON3 and illustrates in detail the coprocessor design targeting G723.1 algorithm acceleration. Facing the problem of pipeline data hazard and operand bits-width that would dramatically reduce instruction performance, different methods are analysed then data-forwarding technology and mirrored register file is adopted. When HW implemented, self-defined coprocessor instructions are employed to optimize the algorithm code and the result is analyzed. In addition, the different memory sub-system configurations are thoroughly experimented on, and the results correspond to theoretical analysis. Finally, G723.1 program is running and tested on FPGA prototype, results of which prove the correctness and effectiveness of coprocessor design.

**Keywords:** LEON3; Coprocessor; G723.1

厦门大学博硕士论文摘要库

# 目录

<b>第 1 章 绪论</b>	<b>1</b>
1.1 研究背景及意义	1
1.1.1 LEON 处理器的研究意义	1
1.1.2 G723.1 语音压缩算法的应用价值	3
1.2 国内外研究现状	4
1.3 论文的内容和结构	5
<b>第 2 章 SOC 的架构和仿真/验证平台</b>	<b>7</b>
2.1 LEON3/GRLIB SOC 平台	7
2.1.1 LEON3/GRLIB 简介	7
2.1.2 AMBA2.0 简介	8
2.2 LEON3 处理器介绍	9
2.2.1 SPARC V8 指令集架构介绍	10
2.2.2 编译器支持	12
2.3 硬件加速器方案	13
2.3.1 紧耦合协处理器和松耦合 IP Core 的比较	14
2.3.2 SIMD 指令的介绍	15
2.4 SOC 原型架构方框图	16
2.5 SOC 仿真平台	17
2.5.1 SOC 的仿真方法	17
2.5.2 TSIM 简介	20
2.5.3 TRAP 介绍	20
2.6 硬件原型验证平台	23
2.6.1 ML509 开发板简介	23
2.6.2 GRMON 简介	24
2.7 小结	25
<b>第 3 章 算法程序性能分析和全局优化</b>	<b>26</b>

<b>3.1</b>	<b>G723.1 算法介绍</b>	<b>26</b>
3.1.1	G723.1 标准及算法原理简介	26
3.1.2	算法实现简介	29
<b>3.2</b>	<b>算法程序性能分析</b>	<b>29</b>
3.2.1	初步性能测量	29
3.2.2	程序性能剖析	31
<b>3.3</b>	<b>全局优化</b>	<b>32</b>
3.3.1	编译优化	32
3.3.2	inline 优化	33
<b>3.4</b>	<b>小结</b>	<b>33</b>
<b>第 4 章</b>	<b>协处理器的设计与实现</b>	<b>34</b>
<b>4.1</b>	<b>协处理器指令集</b>	<b>34</b>
<b>4.2</b>	<b>协处理器微架构设计</b>	<b>36</b>
4.2.1	LEON3 IU 流水线结构分析	36
4.2.2	LEON3 协处理器接口分析	38
4.2.3	协处理器的流水线设计	40
<b>4.3</b>	<b>流水线冲突的解决方法</b>	<b>43</b>
4.3.1	流水线冲突的类型和常用解决方法	43
4.3.2	数据前馈的实现	46
4.3.3	数据前馈的代价和折中	47
<b>4.4</b>	<b>镜像寄存器文件技术</b>	<b>49</b>
<b>4.5</b>	<b>ALU 的流水线设计</b>	<b>50</b>
<b>4.6</b>	<b>实现和仿真</b>	<b>52</b>
4.6.1	VHDL 编码	54
4.6.2	与工艺无关的算术逻辑实现	55
4.6.3	Testbench 及仿真结果	57
4.6.4	综合结果	59
<b>4.7</b>	<b>使用协处理器指令对算法程序进行优化</b>	<b>61</b>
4.7.1	C 语言内嵌汇编语句	61

4.7.2	基本运算函数的改写.....	62
4.7.3	对循环体的优化.....	63
4.8	小结.....	65
<b>第 5 章</b>	<b>SOC 的配置.....</b>	<b>67</b>
5.1	GRLIB 的配置方法.....	67
5.2	IU 的配置 .....	67
5.2.1	寄存器窗数目 .....	67
5.2.2	乘法器配置.....	68
5.3	存储子系统的配置分析 .....	68
5.3.1	层次化的存储子系统.....	68
5.3.2	内存子系统配置.....	71
5.3.3	不同存储方案的实验比较.....	72
5.4	SOC 其它模块的配置 .....	75
5.5	小结.....	75
<b>第 6 章</b>	<b>结论与展望.....</b>	<b>76</b>
6.1	整体结果和分析.....	76
6.1.1	时序性能与硬件资源使用情况.....	76
6.1.2	算法程序运行性能结果及分析.....	77
6.1.3	结论.....	78
6.2	本文工作成果总结 .....	78
6.3	未来工作展望.....	79
	参考文献.....	81
	致 谢 .....	84

厦门大学博硕士论文摘要库

# Contents

<b>Chapter1</b>	<b>Introduction .....</b>	<b>1</b>
1.1	Research Background and Motivation.....	1
1.1.1	Research Motivation for LEON3 processor .....	1
1.1.2	Application Value of G723.1 Vocoder .....	3
1.2	Research Status at Home and Abroad.....	4
1.3	Overview and Structure of the Thesis .....	5
<b>Chapter2</b>	<b>SOC Architecture and Simulation/Verification Platform. 7</b>	
2.1	LEON3/GRLIB SOC Platform Introduction.....	7
2.1.1	LEON3/GRLIB.....	7
2.1.2	AMBA2.0.....	8
2.2	LEON3 Processor Introduction .....	9
2.2.1	SPARC V8 ISA.....	10
2.2.2	Compiler Support.....	12
2.3	HW Accelerator Schemes .....	13
2.3.1	Tightly-Coupled Coprocessor and Loosely-Coupled IP Core .....	14
2.3.2	SIMD instruction .....	15
2.4	SOC Prototype Architecture Block Diagram .....	16
2.5	SOC Simulation Platform .....	17
2.5.1	SOC Simulation Methodologies .....	17
2.5.2	Brief Introduction of TSIM .....	20
2.5.3	TRAP Introduction.....	20
2.6	HW-Prototype Verification Platform .....	23
2.6.1	Brief Introduction of ML509-EVB .....	23
2.6.2	Brief Introduction of GRMON .....	24
2.7	Brief Summary .....	25
<b>Chapter3</b>	<b>Algorithm Profiling and Global Optimization .....</b>	<b>26</b>

<b>3.1</b>	<b>G723.1 Introduction .....</b>	<b>26</b>
3.1.1	G723.1 Codec Standard and Algorithm Principle .....	26
3.1.2	Algorithm Implementation.....	29
<b>3.2</b>	<b>Algorithm Measuring and Profiling .....</b>	<b>29</b>
3.2.1	Performance Measuring .....	29
3.2.2	Algorithm Profiling .....	31
<b>3.3</b>	<b>Global Optimization .....</b>	<b>32</b>
3.3.1	Compilation Optimization .....	32
3.3.2	Inline Optimization .....	33
<b>3.4</b>	<b>Brief Summary .....</b>	<b>33</b>
<b>Chapter4</b>	<b>Coprocessor Design and Implementation .....</b>	<b>34</b>
<b>4.1</b>	<b>Instructions Definition .....</b>	<b>34</b>
<b>4.2</b>	<b>Micro-Architecture Design.....</b>	<b>36</b>
4.2.1	LEON3 IU Pipeline Analysis .....	36
4.2.2	LEON3 Coprocessor Interface Analysis .....	38
4.2.3	Coprocessor Pipeline Design .....	40
<b>4.3</b>	<b>Pipeline Hazard Solution.....</b>	<b>43</b>
4.3.1	Pipeline Hazard Overview and Solutions .....	43
4.3.2	Data-Forwarding Employment .....	46
4.3.3	Cost and Compromise of Data-Forwarding .....	47
<b>4.4</b>	<b>Mirrored Register File .....</b>	<b>49</b>
<b>4.5</b>	<b>Pipelined Design of ALU .....</b>	<b>50</b>
<b>4.6</b>	<b>Implementation and Simulation .....</b>	<b>52</b>
4.6.1	VHDL Coding.....	54
4.6.2	Technology-Independent Implementation of Arithmetic Logic .....	55
4.6.3	Testbench and Simulation Result .....	57
4.6.4	Synthesis Result .....	59
<b>4.7</b>	<b>Algorithm Code Optimization by Coprocessor Instructions .....</b>	<b>61</b>
4.7.1	C Inline Assembly .....	61



Degree papers are in the "[Xiamen University Electronic Theses and Dissertations Database](#)". Full texts are available in the following ways:

1. If your library is a CALIS member libraries, please log on <http://etd.calis.edu.cn/> and submit requests online, or consult the interlibrary loan department in your library.
2. For users of non-CALIS member libraries, please mail to [etd@xmu.edu.cn](mailto:etd@xmu.edu.cn) for delivery details.

厦门大学博硕士论文摘要库